IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	Ashley et al.
Serial Number:	10/621,934
Filing Date:	July 17, 2003
Art Unit:	2452
Examiner:	Hussain, Tauqir
Confirmation Number:	3072
For:	**Method and system for providing user control over receipt of cookies from e-commerce applications**

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

This Brief is submitted pursuant to 37 CFR 41.37.

(i) Real Party In Interest. The real party in interest on this appeal is International Business Machines Corporation ("IBM"), the assignee of record.

(ii) Related Appeals and Interferences. There are no prior and pending appeals, judicial proceedings or interferences known to the appellant that may be related to, directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(iii) Status of Claims. The status of all the claims in the application is set forth in the following claim listing. Each of claims 1-3, 5-12, 14-21 and 23-27 is on appeal.

1. (rejected)
2. (rejected)
3. (rejected)
4. (cancelled)
5. (rejected)
6. (rejected)
7. (rejected)
8. (rejected)
9. (rejected)
10. (rejected)
11. (rejected)
12. (rejected)
13. (cancelled)
14. (rejected)
15. (rejected)
16. (rejected)
17. (rejected)
18. (rejected)
19. (rejected)
20. (rejected)

21. (rejected)

22. (cancelled)

23. (rejected)

24. (rejected)

25. (rejected)

26. (rejected)

27. (rejected)

(iv) Status of Amendments.

An amendment after final under 37 CFR §1.116 was submitted on May 27, 2009. An Advisory Action was mailed June 3, 2009. The claim amendments submitted on May 27th were entered. Thus, this appeal is proceeding on the basis of the claims as of that date.

(v) Summary of Claimed Subject Matter.

The following is a concise explanation of the subject matter defined in each of the independent claims that are the subject of the appeal.

Claim 1 describes a method for processing at a proxy server data transmitted between a server and a client that is operated by a user, wherein the proxy server (page 9, line 28 through page 11, line 8; reference numeral 200 in FIG. 2) communicates with the client (e.g., 202) and the server (e.g., 204) through a network. The method begins by receiving a set of parameters in a client message at the proxy server, wherein the set of parameters are configured by the user at the client. (FIG.s 4A-4C; page 18, line 17 through page 20, line 22; page 19, lines 9-13) The set of parameters is stored at the proxy server. (FIG. 2, active user entry 210; domain lists 220 and 222; page 15, lines 2-17) The parameters comprise domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers. (FIG. 2, active user entry 210; see also page 20, lines 5-10) The method then continues by receiving at the proxy server a response message from the server for the client. (FIG. 5, step 502, page 20, lines 26-28) The method detects at the proxy server a cookie associated with the response message. (FIG. 5, step 506, page 21, lines 2-6) Thereafter, the method continues by extracting from the response message a domain identifier associated with the server. (FIG. 5, step 510 generally; page 21, lines 6-10) The method then continues by retrieving the set of parameters. (FIG. 6A,

steps 602 and 604; page 22, lines 1-17). The cookie is then processed at the proxy server in accordance with the retrieved set of parameters and the extracted domain identifier. (See generally FIG. 6A, steps 614, 616, 618 and 620; page 22, line 1 through page 24, line 7)

Claim 10 describes an apparatus for processing at a proxy server data transmitted between a server and a client that is operated by a user, wherein the proxy server (page 9, line 28 through page 11, line 8; reference numeral 200 in FIG. 2) communicates with the client and the server through a network. The apparatus comprises a processor (FIG. 1B, numeral 122), and a computer memory (FIG. 1B, numerals 124, 126 and/or 132) holding computer program instructions (computer software in reference numeral 200) which when executed by the processor perform a method like that described in claim 1. The method begins by receiving a set of parameters in a client message at the proxy server, wherein the set of parameters are configured by the user at the client. (FIG.s 4A-4C; page 18, line 17 through page 20, line 22; page 19, lines 9-13) The set of parameters is stored at the proxy server. (FIG. 2, active user entry 210; domain lists 220 and 222; page 15, lines 2-17) The parameters comprise domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers. (FIG. 2, active user entry 210; see also page 20, lines 5-10) The method then continues by receiving at the proxy server a response message from the server for the client. (FIG. 5, step 502, page 20, lines 26-28) The method detects at the proxy server a cookie associated with the response message. (FIG. 5, step 506, page 21, lines 2-6) Thereafter, the method continues by extracting from the response message a domain identifier associated with the server. (FIG. 5, step 510 generally; page 21, lines 6-10) The method then continues by retrieving the set of parameters. (FIG. 6A, steps 602 and 604; page 22, lines 1-17). The cookie

is then processed at the proxy server in accordance with the retrieved set of parameters and the extracted domain identifier. (See generally FIG. 6A, steps 614, 616, 618 and 620; page 22, line 1 through page 24, line 7)

Claim 19 describes a computer program product in a computer readable medium (page 26, line 31 through page 27, line 2) for use at a proxy server for processing data transmitted between a server and a client that is operated by a user, wherein the proxy server (page 9, line 28 through page 11, line 8; reference numeral 200 in FIG. 2) communicates with the client and the server through a network, the computer program product holding computer program instructions which when executed by the proxy server perform a method like that described in claim 1. The method begins by receiving a set of parameters in a client message at the proxy server, wherein the set of parameters are configured by the user at the client. (FIG.s 4A-4C; page 18, line 17 through page 20, line 22; page 19, lines 9-13) The set of parameters is stored at the proxy server. (FIG. 2, active user entry 210; domain lists 220 and 222; page 15, lines 2-17) The parameters comprise domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers. (FIG. 2, active user entry 210; see also page 20, lines 5-10) The method then continues by receiving at the proxy server a response message from the server for the client. (FIG. 5, step 502, page 20, lines 26-28) The method detects at the proxy server a cookie associated with the response message. (FIG. 5, step 506, page 21, lines 2-6) Thereafter, the method continues by extracting from the response message a domain identifier associated with the server. (FIG. 5, step 510 generally; page 21, lines 6-10) The method then continues by retrieving the set of parameters. (FIG. 6A, steps 602 and 604; page 22, lines 1-17). The cookie is then processed at the proxy server in accordance

with the retrieved set of parameters and the extracted domain identifier. (See generally FIG. 6A, steps 614, 616, 618 and 620; page 22, line 1 through page 24, line 7)

Means-plus-function (MPF) structure

There are no MPF-style claim limitations in the pending claims.

(vi) Grounds of Rejection to be Reviewed.

Group I – Claims 1, 7-10, 16-19 and 25-27

Whether the Examiner erred in finding that Nilsson et al, WO 99/64967 ("Nilsson"), in view of Internet Request for Comment 2965 (the "RFC"), is the subject matter, taken as a whole, of any of claims 1, 7-10, 16-19 and 25-27?

Group II – Claims 2, 11 and 20

Whether the Examiner erred in finding that Nilsson, in view of the RFC, is the subject matter, taken as a whole, of any of claims 2, 11, and 20?

Group III – Claims 3, 12 and 21

Whether the Examiner erred in finding that Nilsson, in view of the RFC, is the subject matter, taken as a whole, of any of claims 3, 12, and 21?

Group IV – Claims 5, 14 and 23

Whether the Examiner erred in finding that Nilsson, in view of the RFC, is the subject matter, taken as a whole, of any of claims 5, 14, and 23?

Group V - Claims 6, 15 and 24

Whether the Examiner erred in finding that Nilsson, in view of the RFC, is the subject matter, taken as a whole, of any of claims 6, 15, and 24?

(vii) Argument.

All pending claims stand rejected on obviousness grounds in view of the Nilsson-RFC combination.

The applicable legal principles are straightforward. "[A]n applicant can overcome a [Section 103] rejection by showing insufficient evidence of *prima facie* obviousness or by rebutting the *prima facie* case, ..." *See*, *In re Kahn*, 441 F.3d 977, 985-86 (Fed. Cir. 2006). In considering this question, it is appropriate to consider whether the Examiner has met his or her burden to show that the claimed subject matter is disclosed "clearly and unequivocally" in a cited reference. *In re Arkley*, 455 F.2d 586, 587 (CCPA 1972). This issue is evaluated from the viewpoint of a person of ordinary skill in the art, who is a person of ordinary creativity, not an automaton. *KSR Int'l Co. v. Teleflex, Inc.*, 55o U.S. 398, 421, 127 S.Ct. 1727 (2007).

Whether or not particular subject matter "as a whole" would have been obvious to one of ordinary skill in the art at the time an invention was made depends on underlying factual inquiries including: (1) the scope and content of the prior art; (2) the level of ordinary skill in the art; and (3) the differences between the prior art and the claimed invention. *Graham v. John Deere Co.*, 383 U.S. 1, 17 (1966). In rejecting claims under 35 U.S.C. § 103(a), it is incumbent upon the Examiner to establish a factual basis to support the legal conclusion of obviousness. *See*, *In re Fine*, 837 F.2d 1071, 1073 (Fed. Cir. 1988).

Under *KSR Int'l v. Teleflex, Inc.*, an initial inquiry is whether the claimed invention can be fairly characterized as involving a simple substitution of one known element for another or the mere application of a known technique to a piece of prior art allegedly ready for the improvement. If the invention cannot be so characterized, obviousness cannot be made out

unless the Office can establish "some articulated reasoning with some rational underpinning" – *viz.*, an apparent reason to combine the known elements in the fashion claimed. See, *KSR Int'l v. Teleflex, Inc.*, 127 S Ct. 1727, 1740-41 (2007). The articulated reasoning can be based on interrelated teachings of multiple patents, market demand, or the background knowledge of one of ordinary skill.

One cannot show non-obviousness by attacking references individually where the rejection is based on a combination of the references, *In re Keller*, 642 F.2d 413, 416 (CCPA 1981).

In considering grounds of rejection, "every limitation in the claim must be given effect rather than considering one in isolation from the others." See, *In re Geerdes*, 491 F. 2d 1260, 1262-63 (CCPA 1974). Moreover, a rejection based on prior art cannot be based on speculations and assumptions. *In re Steele*, 305 F. 2d 859, 862 (CCPA 1962). Further, every limitation in a claim is material to patentability. (*See*, 35 U.S.C. §103(a) concerning the subject matter "as a whole").

Group I – Claims 1, 7-10, 16-19 and 25-27

By way of brief background, the subject matter of this application relates generally to a privacy proxy server or privacy service. If a user of such a system or service is very mobile and uses many different client devices, there may be occasions or environments in which the user would like to receive some or all "cookies" (typically, HTTP files produced by web servers) that include tracking information at a client device while filtering out some or all cookies in a different environment or on a different occasion, even though the user may or may not continue to employ a privacy proxy or privacy service in these different environments or upon these

different occasions. For example, if a user only accesses a certain web site from the user's personal laptop and never from an office desktop, then the user may want to allow cookies through the privacy proxy server to the laptop; the laptop would tend to have the latest cookies stored in its cookie cache, which might be important for certain sites that are highly customized or individualized. In this example, the user's laptop would have recent cookies if the user decided to use the laptop without accessing the Web through the privacy proxy server.

With the subject matter described herein, the user is able to create different client profiles based on the user's needs, thereby giving the user a finer granularity of control over the cookie filtering functionality of a privacy proxy server or a privacy service. In particular, with the described subject matter, the user can *customize* the operation of the privacy proxy server or the privacy service on the basis of *user-configured* information. The privacy proxy then filters cookies that are returned by the server *in accordance with user-configurable parameters.*

The "user-configurability" subject matter is incorporated into each independent claim 1, 10 and 19.

In particular, each independent claim positively recites the steps of receiving and storing a set of parameters, wherein the parameters comprise "domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers." As positively recited, the set of parameters are configured by the user at the client.

Turning now to the rejection, the first issue concerns the "scope and content" of each prior art reference.

Nilsson illustrates a proxy server 66 located between a mobile device user terminal 52 having a browser 54, and a server 70. In Nilsson, the goal is to intercept and store a cookie

generated by the server 70 so that the user terminal 52 does not need to store the cookie. In operation, the client makes a request to the server, which then provides a response together with the cookie. This is a conventional server operation. Rather than passing the cookie back to the user terminal, the cookie is stored in the proxy together with information identifying the requested URL and the user terminal. In that manner, "the next time" the user terminal 52 accesses the server 70, the proxy server 66 matches the requested URL and the identification information and in this manner finds the previously-stored cookie. The cookie is then provided to the server 70 with the request. Thus, the cookie is stored in the proxy server 66 and need not be transmitted over a wireless interface.

In the "Response to Arguments" section in the final rejection, the Examiner contends for the first time that, because Nilsson discloses a client and proxy connected for communication purposes, such a communication "will require initialization parameters between client and proxy since at this point there is no specific Message instruction having any specific parameters and therefore, any parameter can be read as [the claimed] "set of parameters" which has to be configured on [the] user computer to communicate with proxy." (See, Final rejection at page 2) The Examiner further contends that the Nilsson abstract's discussion of cookie intercept, storage and processing necessarily means that the reference teaches the claimed "extracting from the response message a domain identifier associated with the server" and the "processing the cookie at the proxy server in accordance with the retrieved set of parameters and the extracted domain identifier." (See, Final rejection at page 3) The Examiner does admit, however, that Nilsson does not disclose the "specifics of [the] Cookies parameter."

The RFC is cited for its alleged teaching of a "set of parameters," with reference in particular to paragraph 3.3.4. This section of the RFC describes "Sending Cookies to the Origin Server" and is a description of how a client user agent (a web browser) may include a cookie request header if it has stored cookies that are applicable to the request. According to the RFC, the cookie request header may be based on the request-host and request-port, the request URL, and the cookie's age. The syntax for the cookie request header is described and includes various attributes and values.

In the "Response to Arguments" section in the final rejection, the Examiner cited (for the first time) Section 3.2, which mentions a "Secure" attribute that can be determined "possibly with user interaction," as well as Section 3.3.3, which purports to describe further "customization of parameters in detail."

Respectfully, the obviousness rejection is traversed.

With respect, the Examiner still has erred in applying Nilsson to the claimed subject matter.

In particular, the Examiner argues (at the "Response to Arguments") that Nilsson performs the step of "extracting from the response message a domain identifier associated with the server." This is incorrect; what actually happens at the Nilsson proxy server 66 is the exact opposite step, because at this point in the Nilsson processing the proxy 66 operates on the request message from the end user terminal and not the response message from the server 70. This distinction is clear from the very portion of the Nilsson text originally relied upon by the Examiner, namely, the text at page 3, lines 9-11 (emphasis supplied):

"Thus, when a remote HTTP server or the like is contacted by a user terminal and the remote server transmits a cookie to the user terminal, the cookie is intercepted and stored in the

proxy server. Information regarding the remote server, e.g., its URL[,] and an identification identifying the user terminal or the user is stored together with the cookie. The next time the user terminal or the user accesses the same HTTP server the proxy server matches the requested URL and the identification information and in this manner finds the stored cookie. The cookie is then added to the request message so that the remote server is accessed with a copy of the cookie as desired."

The Nilsson abstract, which the Examiner relies upon in the final rejection, says exactly the same thing, as it references the "next time" wording as well on line 5. Importantly, at this stage in the Nilsson operation, the "domain identifier" – if any – is in the URL in the request message, as opposed to the domain identifier in any server response message. Thus, the portion of the text relied upon by the Examiner does not meet the claim limitation itself.

Moreover, the final step of claim 1 requires "processing the cookie at the proxy server in accordance with the retrieved set of parameters and the extracted domain identifier." As noted in the previous paragraph, the "extracted domain identifier" referenced at this part of the claim refers to the prior "extracting" step where the "domain identifier" is extracted from the server response message; as noted above, the URL obtained by the proxy at this point in the Nilsson operation is an identifier taken from the end user request message itself. In other words, in the Nilsson operation (and after the cookie is stored at the proxy), when the end user terminal issues a next request to the server, the proxy determines whether there is a stored cookie and, if so, the proxy merely attaches the cookie to the request before passing the request to the server. The cookie itself is not processed (as required by claim 1, 10 and 19) "in accordance with [any] retrieved set of parameters and the extracted domain identifier." Stated another way, in Nilsson at most what happens is that the cookie is re-attached to the end user terminal request before that request is passed on to the server.

Further, the "set of parameters" referenced in the claim are parameters that "are configured by the user at the client." In the Office action mailed August 5, 2008, the Examiner argued that this limitation was met by the RFC. In response to that initial position, the undersigned argued (in the Response submitted November 5, 2008) that the Examiner was incorrect, that this feature was not taught in the RFC. The Examiner responds now that, in effect, the feature is actually present in both Nilsson and the RFC. This is not the case.

First, with respect to Nilsson, the Examiner provides no textual or other support in Nilsson for the conclusion that "since at this point there is no specific Message instruction having any specific parameters[;] therefore, any parameter can be read as [the claimed] "set of parameters" which has to be configured on [the] user computer to communicate with proxy." (See, Final rejection at page 2). As noted above, a rejection based on prior art cannot be based on speculations and assumptions. *In re Steele*, 305 F. 2d at 862. More to the point, the Examiner's argument is circular and misses the important point that the "set of parameters" referenced in the claim are parameters that "are configured by the user." While the Examiner's argument and rationale are not entirely clear, he seems to be arguing that the fact that the client is in communication with the proxy necessarily means that something on the client is passing a parameter to the proxy; thus, according to the Examiner, the claim limitation is met. This argument rewrites the claim language and, for that reason, is improper. This is not a question of affording the claim phrase its broadest reasonable interpretation and then reading that claim on the reference (see, MPEP §2111); rather, here the Examiner has simply misread what the claim actually says; the "parameters" must be "configured by the user at the client." This is not simply saying that something in the client sends some unspecified data to the proxy server to facilitate a

communication. The Examiner is simply reading the claim limitation without the "configured by the user" clause. This is error.

Turning to the RFC, the Examiner newly cites Section 3.2, which recites a "Secure" attribute that may be included in a cookie response header. While this portion of the RFC does state that the attribute may be determined "possibly with user interaction," this attribute is not the specified "set of parameters" that the claim itself defines: "wherein the parameters comprise domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers." According to the RFC, the "Secure attribute should be considered security advice from the server to the user agent, indicating that it is in the session's interest to protect the cookie contents." At most then, what the RFC teaches one of ordinary skill is that the Secure attribute is a designation of a "level of security." According to the RFC, if there is "no value" designated and the attribute is set, the attribute "directs the user agent to use only (unspecified) secure means to contact the origin server whenever it sends back the cookie." Otherwise, there is some "level of security" identified. In no event, however, is the Secure attribute the specified "set of parameters" as that term is actually defined in claim 1, 10 and 19.

Neither Nilsson nor the RFC disclose or suggest user-configurable parameters that comprise "domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers." Further, neither Nilsson nor the RFC disclose proxy server processing of a cookie (and, in particular, a "domain identifier") that is being returned from a server to a client, let alone per-domain cookie filtering "in accordance with the retrieved set of parameters and [an] extracted domain identifier." Nilsson has nothing to do with cookie processing; rather, the entire point of that scheme is to store cookies at the proxy so that

they do not have to be returned to the requested end user terminals. Stated another way, in Nilsson there is no cookie processing function because the whole purpose of the proxy is to store the server-generated cookies so that they can be re-used. In other words, there is no function in Nilsson that "process[es] the cookie." In particular, an end user of the terminal 52 cannot configure the proxy 66 in any way, let alone to allow some cookies to pass through while others are merely stored.

The Examiner contends that the RFC teaches the "processing the cookie" limitation and points to Section 3.3.6 where it is alleged that an "unverifiable transaction" blocks the cookie from transmission, as well as Section 3.2.3 that discusses cookie caching such "a modified response" is sent to the client. (See, Final rejection at pages 3-4) With respect, once again the Examiner is not reading the claim limitation in its entirety. In particular, Section 3.3.6 concerning "unverifiable transactions" involves a function at the "user agent" – namely, the client – and concerns whether the client should or should not send the cookie back to the origin server that provided it in the first instance. The claim limitation, in contrast, refers to "processing the cookie at the proxy server." The "unverifiable transaction" processing described in the RFC is not the same thing and, more to the point, the claim limitation refers to "processing the cookie at the proxy server in accordance with the retrieved set of parameters and the [] domain identifier" extracted during the "response message" processing (at the proxy server). Section 3.3.6 just concerns whether the user agent will send the cookie back to the origin server. The claim language is not met by this teaching.

The "cache control" directives in Section 3.2.3 are not the claimed subject matter either. This section of the RFC simply states that the origin server can include such directives to control

the downstream caching of the associated content (the "returned resource") in question. The "processing the cookie" that is being done in the claimed subject matter is done "in accordance with the retrieved set of parameters (in other words, those stored at the proxy server and as "configured by the user") and the domain identifier extracted during the "response message" processing (at the proxy server). The cache control directive in Section 3.2.3 itself is nothing more than an instruction about whether to cache the associated cache. At most, the cache control directive is an instruction that is returned in the response message, but this is not a "processing [of] the cookie" itself, let alone in the manner positively recited in the claim.

Nilsson simply teaches cookie storage in the proxy to obviate passing the cookie back to the requesting end user terminal, and then later re-using the cookie to obtain access to a resource on the server. This is not the subject matter of the claims here. Rather, claims 1, 10 and 19 here assume that the client has obtained access to the server and that the server has issued the cookie. Unlike the cited art, the claims concern whether that cookie will be returned to the client. As noted above, no cookies are ever returned to the requesting end user terminals 52 in the Nilsson scheme. This cookie processing concept is not disclosed or suggested by any of the art of record, as none of the references even address the question of how a cookie being returned from a server to the client should be processed, let alone processed according to user-configurable options. The RFC simply discloses conventional cookie management in an Internet-compliant web browser (user agent) and server interaction.

A test for obviousness is what the combined teachings of the references would have suggested to those of ordinary skill in the art, *In re Keller*, 642 F.2d 413, 426 (CCPA 1981). 426. Here, the "combined teachings" of Nilsson/RFC simply describe a system of cookie storage

in a proxy to obviate passing the cookie back to a requesting end user terminal, and then later re-using the cookie to obtain access to a resource on the server. The cookie may include a Security attribute that identifies a level of security that should be imposed by the user agent when the cookie is re-used, and that security level may be determined "possibly with user interaction." The cookie may be returned to the proxy server with a cache control directive associated with the resource that was requested by the client.

The Nilsson/RFC combination does not describe providing a technique for enabling a user to configure at the proxy server per-domain filtering of cookies that are returned from servers. Rather, Nilsson, the primary reference, deals with an unrelated issue, *viz.*, how to store and re-use a cookie so that the requesting end user terminal does not need to store it directly. The RFC does not make up for the deficiencies in the primary reference for the reasons advanced above.

In particular, the combination of the cited art still fails to disclose at least the following steps of claims 1, 10 and 19:

"receiving a set of parameters in a client message at the proxy server, wherein the set of parameters are configured by the user at the client;

storing the set of parameters at the proxy server, wherein the parameters comprise domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers;

extracting from the response message a domain identifier associated with the server;

retrieving the set of parameters; and

processing the cookie at the proxy server in accordance with the retrieved set of parameters and the extracted domain identifier."

In considering alleged obviousness, each limitation is material to patentability, as Section 103 requires that the subject matter "as a whole" be shown in the recited combination.

This user-configurable per-source domain "processing" provides an enhanced privacy service that is neither disclosed nor suggested by the prior art. Thus, independent claims 1 (method), 10 (apparatus) and 19 (computer program product) describe patentable subject matter.

Dependent claims 7-9, 16-18 and 25-27 are patentable for the reasons advanced with respect to the parent claims. These claims are not being separately argued.

For the above reasons, the rejection of claims 1, 7-10, 16-19 and 25-27 should be withdrawn.

Group II – Dependent Claims 2, 11 and 20

Dependent claims 2, 11 and 20 describe the cookie processing steps more specifically and, in particular, the steps of blocking the cookie from transmission, caching the cookie at the proxy, and sending a modified response message to the client. This is the scenario such as described in steps 614, 618 and 620 of FIG. 6A, where the user has selected an option not to allow the cookie through the privacy service proxy server.

Here, the Examiner contends (in the "Response to Arguments" at page 3-4) that Section 3.3.6 discloses the "unverifiable transaction" function that is said to disclose "blocking the cookie from transmission" as well as Section 3.2.3 directed to controlling "caching." These contentions, as noted above, are overstated. As noted above, the "unverifiable transaction" function is an operation carried out at and by the user agent (the client) - not the proxy server -

and simply involves the question of whether the user agent will send the cookie back to the server (in a subsequent transaction). More importantly, in rejecting these claims, the Examiner has not fairly considered the conditional clause, namely, that the cookie blocking is carried out "in response to a determination that the set of parameters contains the extracted domain identifier." As noted above, neither Nilsson nor the RFC disclose or suggest making a determination that compares the "set of [user-configured] parameters" with the "domain identifier" extracted from the server "response message." Thus, the specific "in response to" operation cannot be (and is not) found in the Nilsson/RFC combination.

As noted above, Nilsson has no concept of selectively blocking some cookies while allowing others to pass back through to the client. RFC Section 3.3.6 simply concerns whether the user agent will send the cookie back. These claims are separately patentable because the cited art does not teach any filtering of cookies being returned from a server to a client, let alone the specific requirements set forth in these claims.

Group III – Dependent Claims 3, 12 and 21

Dependent claims 3, 12 and 21 likewise describe the cookie processing steps but in this case describe the operation where the cookie (of a recognized domain) is passed back to the client. This is the scenario such as described in step 614 and 616 of FIG. 6A, where the user has selected an option to allow the cookie through the privacy service, in which case the privacy service sends the response to the client without removing or detaching the cookie from the response.

As described above, Nilsson teaches away from this requirement, *as cookies are retained at and by the proxy.*

Discussing the question of obviousness of a patent that claims a combination of known elements, *KSR Int'l Co. v. Teleflex, Inc.*, 550 U.S. 398, 127 S.Ct. 1727 (2007), explains "when the prior art teaches away from combining certain known elements, discovery of a successful means of combining them is more likely to be nonobvious." Id. at 1740 (citing United States v. Adams, 383 U.S. 39, 51-51 (1966). Additionally, "[a] reference may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant." *In re Kahn*, 441 F.3d 977, 990 (Fed. Cir. 2006) (quoting *In re Gurley*, 27 F.3d 551, 553 (Fed. Cir. 1994)). To "teach away," a reference typically should warn a person of ordinary skill in the art not to apply a certain structure or technique. *Para-Ordnance Mfg., Inc. v. SGS Importers Int'l, Inc.*, 73 F3d 1085, 1090 (Fed. Cir. 1995). See also, *In re Gordon*, 733 F. 2d 900 (Fed.Cir. 1984) (a proposed combination is improper if it renders a reference inoperable for its intended purpose).

Here, one of ordinary skill in the art would never modify Nilsson to pass the cookie back to the client, as the entire Nilsson system is designed to avoid precisely that function. The reference specifically teaches "by storing cookie information in a proxy server, the cookies do not need to be stored in the user terminal, which in many cases have a small memory and which therefore is not suited for storing cookies." (See, page 9). Further, "when the user terminal is a mobile terminal the cookie is not transmitted over an air-interface [back to the client], thereby reducing the amount of transmitted over the air interface significantly." (See, page 3)

The invention of claims 3, 12 and 21 cannot be fairly characterized as involving a simple substitution of one known element (the RFC cookie specification) for another (the Nilsson proxy

server cookie striping and storage function) or the mere application of a known technique to a piece of prior art (the Nilsson proxy) allegedly ready for the improvement. Moreover, because the invention cannot be so characterized, obviousness cannot be made out unless the Office can establish "some articulated reasoning with some rational underpinning" – *viz.*, an apparent reason to combine the known elements in the fashion claimed. See, *KSR Int'l v. Teleflex, Inc.*, 127 S Ct. at 1740-41. As noted above, while the articulated reasoning can be based on interrelated teachings of multiple patents, market demand, or the background knowledge of one of ordinary skill, it would not be "rational" to modify Nilsson here because, as noted above, the reference teaches away from the explicit claim requirement of "sending the response message along with its associated cookie to the client."

These claims also include a conditional clause, namely, that the sending the response message function takes place "in response to a determination that the set of parameters contains the extracted domain identifier." As noted above, neither Nilsson nor the RFC disclose or suggest making a determination that compares the "set of [user-configured] parameters" with the "domain identifier" extracted from the server "response message." Thus, the specific "in response to" operation cannot be (and is not) found in the Nilsson/RFC combination.

For all these reasons, claims 3, 12 and 21 are separately patentable over the combination.

Group IV – Dependent Claims 5, 14 and 23

Dependent claims 5, 14 and 23 are separately patentable as they describe the further step of determining if the set of parameters contains an indication that the user has enabled cookie processing by the proxy server. In one embodiment, this refers to determining whether a "source

domain filter enable flag" (218) is set. The cited art does not perform cookie processing, so this functionality is also absent from the Nilsson/RFC combination.

Group V – Dependent Claims 6, 15 and 24

Dependent claims 6, 15 and 24 are separately patentable as they describe the further steps of managing "multiple sets of parameters for the user." This is a client profile option.

As noted above, the "set of parameters" referenced in the claim are parameters that "are configured by the user at the client."

With respect to Nilsson and as argued above, the Examiner provides no textual or other support in Nilsson for the conclusion that "any [Nilsson] parameter can be read as [the claimed] "set of parameters" which has to be configured on [the] user computer to communicate with proxy." (See, Final rejection at page 2). As noted above, a rejection based on prior art cannot be based on speculations and assumptions. *In re Steele*, 305 F. 2d at 862. The "set of parameters" referenced in the claim are parameters that "are configured by the user" and this set of dependent claims further requires "multiple" such "sets of parameters." Nilsson does not show this element.

The RFC "Secure" attribute is not the specified "set of parameters" that the claim itself defines: "wherein the parameters comprise domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers." At most, what the RFC teaches one of ordinary skill is that the Secure attribute is a designation of a "level of security." According to the RFC, as noted above if there is "no value" designated and the attribute is set, the attribute "directs the user agent to use only (unspecified) secure means to contact the origin server whenever it sends back the cookie." Otherwise, there is some "level of

security" identified. In no event, however, is the Secure attribute the specified "set of parameters" as that term is actually defined in the parent claims, let alone the "multiple" such set required here.

Neither Nilsson nor the RFC disclose or suggest "multiple" user-configurable "sets of parameters" that comprise "domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers."

With respect to these limitations, there is "insufficient evidence of *prima facie* obviousness." *See*, *In re Kahn*, 441 F.3d at 985-86.

For the reasons stated above, the obviousness rejections are in error and should be withdrawn.

Respectfully submitted,

/David H. Judson/

By: _____

David H. Judson, Reg. No. 30,467

July 27, 2009

(viii) Claims Appendix.

1. (previously presented) A method for processing at a proxy server data transmitted between a server and a client that is operated by a user, wherein the proxy server communicates with the client and the server through a network, the method comprising:

receiving a set of parameters in a client message at the proxy server, wherein the set of parameters are configured by the user at the client;

storing the set of parameters at the proxy server, wherein the parameters comprise domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers;

receiving at the proxy server a response message from the server for the client;

detecting at the proxy server a cookie associated with the response message;

extracting from the response message a domain identifier associated with the server;

retrieving the set of parameters; and

processing the cookie at the proxy server in accordance with the retrieved set of parameters and the extracted domain identifier.

2. (original) The method of claim 1 further comprising:

in response to a determination that the set of parameters contains the extracted domain identifier, blocking the cookie from transmission from the proxy server to the client;

caching the cookie at the proxy server; and

sending a modified response message to the client.

3. (original) The method of claim 1 further comprising:

in response to a determination that the set of parameters contains the extracted domain identifier, sending the response message along with its associated cookie to the client.

4. (cancelled)

5. (original) The method of claim 1 further comprising:

determining, prior to processing the cookie at the proxy server in accordance with the retrieved set of parameters and the extracted domain identifier, if the set of parameters contains an indication that the user has enabled cookie processing by the proxy server.

6. (original) The method of claim 1 further comprising:

managing multiple sets of parameters for the user at the proxy server, wherein each set of parameters is associated with an identifier; and

selecting by the user a first identifier that is associated with the set of parameters prior to retrieving the set of parameters, wherein the set of parameters is retrieved in accordance with the selected first identifier.

7. (original) The method of claim 6 wherein the first identifier is selecting during an authentication operation.

8. (original) The method of claim 6 further comprising:

selecting a second identifier; and

processing the cookie at the proxy server in accordance with a set of parameters that is associated with the second identifier.

9. (original) The method of claim 6 wherein identifiers that are associated with sets of parameters are chosen from a group comprising a type of client device or a client location.

10. (previously presented) An apparatus for processing at a proxy server data transmitted between a server and a client that is operated by a user, wherein the proxy server communicates with the client and the server through a network, the apparatus comprising:

a processor;

a computer memory holding computer program instructions which when executed by the processor perform a method comprising:

receiving a set of parameters in a client message at the proxy server, wherein the set of parameters are configured by the user at the client;

storing the set of parameters at the proxy server, wherein the parameters comprise domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers;

receiving at the proxy server a response message from the server for the client;

detecting at the proxy server a cookie associated with the response message;

extracting from the response message a domain identifier associated with the server;

retrieving the set of parameters; and

processing the cookie at the proxy server in accordance with the retrieved set of parameters and the extracted domain identifier.

11. (currently amended) The apparatus of claim 10 wherein the method further comprises:

blocking the cookie from transmission from the proxy server to the client in response to a determination that the set of parameters contains the extracted domain identifier;

caching the cookie at the proxy server; and

sending a modified response message to the client.

12. (previously presented) The apparatus of claim 10 wherein the method further comprises:

sending the response message along with its associated cookie to the client in response to a determination that the set of parameters contains the extracted domain identifier.

13. (cancelled)

14. (previously presented) The apparatus of claim 10 wherein the method further comprises:

determining, prior to processing the cookie at the proxy server in accordance with the retrieved set of parameters and the extracted domain identifier, if the set of parameters contains an indication that the user has enabled cookie processing by the proxy server.

15. (previously presented) The apparatus of claim 10 wherein the method further comprises:

managing multiple sets of parameters for the user at the proxy server, wherein each set of parameters is associated with an identifier; and

selecting by the user a first identifier that is associated with the set of parameters prior to retrieving the set of parameters, wherein the set of parameters is retrieved in accordance with the selected first identifier.

16. (original) The apparatus of claim 15 wherein the first identifier is selecting during an authentication operation.

17. (previously presented) The apparatus of claim 15 wherein the method further comprises:

selecting a second identifier; and

processing the cookie at the proxy server in accordance with a set of parameters that is associated with the second identifier.

18. (original) The apparatus of claim 15 wherein identifiers that are associated with sets of parameters are chosen from a group comprising a type of client device or a client location.

19. (previously presented) A computer program product in a computer readable medium for use at a proxy server for processing data transmitted between a server and a client that is operated by a user, wherein the proxy server communicates with the client and the server through a network, the computer program product holding computer program instructions which when executed by the proxy server perform a method comprising:

receiving a set of parameters in a client message at the proxy server, wherein the set of parameters are configured by the user at the client;

storing the set of parameters at the proxy server, wherein the parameters comprise domain identifiers associated with indications of whether to block transmission of cookies associated with the domain identifiers;

receiving at the proxy server a response message from the server for the client;

detecting at the proxy server a cookie associated with the response message;

extracting from the response message a domain identifier associated with the server;

retrieving the set of parameters; and

processing the cookie at the proxy server in accordance with the retrieved set of parameters and the extracted domain identifier.

20. (previously presented) The computer program product of claim 19 wherein the method further comprises:

blocking the cookie from transmission from the proxy server to the client in response to a determination that the set of parameters contains the extracted domain identifier;

caching the cookie at the proxy server; and

sending a modified response message to the client.

21. (previously presented) The computer program product of claim 19 wherein the method further comprises:

sending the response message along with its associated cookie to the client in response to a determination that the set of parameters contains the extracted domain identifier.

22. (cancelled)

23. (previously presented) The computer program product of claim 19 wherein the method further comprises:

determining, prior to processing the cookie at the proxy server in accordance with the retrieved set of parameters and the extracted domain identifier, if the set of parameters contains an indication that the user has enabled cookie processing by the proxy server.

24. (previously presented) The computer program product of claim 19 wherein the method further comprises:

managing multiple sets of parameters for the user at the proxy server, wherein each set of parameters is associated with an identifier; and

selecting by the user a first identifier that is associated with the set of parameters prior to retrieving the set of parameters, wherein the set of parameters is retrieved in accordance with the selected first identifier.

25. (original) The computer program product of claim 24 wherein the first identifier is selecting during an authentication operation.

26. (previously presented) The computer program product of claim 24 wherein the method further comprises:

selecting a second identifier; and

processing the cookie at the proxy server in accordance with a set of parameters that is associated with the second identifier.

27. (original) The computer program product of claim 24 wherein identifiers that are associated with sets of parameters are chosen from a group comprising a type of client device or a client location.

(ix) Evidence Appendix.

None.

(x) Related Proceeding Appendix.

None.